ITCS 113 – Fundamentals of Programming
Lecture 4: Extra Exercises

No Need to Submit but Practices (ALL) is encouraged.

1. Write the correct expression of the `do-while` loop to make the program validate the input `x`. If the input does not match the given conditions, the program should continue asking for the input until the value of `x` matches the given condition.

Condition	Loops

x is a negative integer number
e.g. `-256, -9, -1`
	`int x;`
	`do {`
	`  scanf("%d", &x);`
	`} while(  );`
	`printf("Input is %d", x);`

x is an uppercase alphabet
e.g. `'A', 'B', 'C'`
	`char x;`
	`do {`
	`  scanf(" %c", &x);`
	`} while(  );`
	`printf("Input is %c", x)`

x is a positive odd number
e.g. `'5', '3', '201'`
	`int x;`
	`do {`
	`  scanf("%d", &x);`
	`}while(  );`
	`printf("Input is %c", x)`

x is between 50 to 150 inclusively
e.g. `50, 51, …, 150`
	`int x;`
	`do {`
	`  scanf("%d", &x);`
	`}while(  );`
	`printf("Input is %c", x)`

2. Write the following series using Loop
   - `4 5 6 7 8 9 10`
   - `-18 -15 -12 -9 -6 -3 0 3 6 9 12 15 18`
   - `5 10 15 20 25 30 35 40 45 50 55 60`
   - `1 2 4 8 16 32 64`
   - `1A 2B 3A 4B 5A 6B 7A 8B 9A 10B`
   - `50 40 30 20 10 0 -10 -20 -30 -40 -50`

3. Write a program in C to display the n terms of a harmonic series and their sum, i.e., `1 + 1/2 + 1/3 + 1/4 + 1/5 ... 1/n` terms.
   Example: `n=5; 1 + 1/2 + 1/3 + 1/4 + 1/5 = 2.28`

4. Write a C program that repeatedly takes reinput from the user and counts the number of uppercase and lowercase letters, until users input a letter '#'

5. Write a program that receives a positive integer n from users. Print all even numbers those are within a range of -n to n and skip 0 (zero).

6. Write a program that prints English letters from `'a'` to `'z'` with a comma separating them **using LOOP statements**. The program prints the UPPERCASE letters at the **even** position, and the lowercase letters at the **odd** position. `'A'` is the 1st position, `'b'` is the 2nd position, and so on. (Hint: the ASCII code for `'A'` is `65` and `'a'` is `97`)
   Note: There is no comma after the last letter
   Expected output:
   `a,B,c,D,e,F,g,H,i,J,k,L,m,N,o,P,q,R,s,T,u,V,w,X,y,Z`

7. Write a C program that repeatedly takes reinput from the user and counts the number of uppercase and lowercase letters, until users input a letter '#'

8. Write a C program to similate the 7-up games. The program will display the series of numbers from 1 to `n`. If the number is ending with 7 (e.g., 7, 17, 27, …) or the number is divisible by 7 (e.g., 7, 14, 21, …). The program will skip the number and show the subsequent number until `n`
   Example: `n=20; 1 2 3 4 5 6 8 9 10 11 12 13 15 16 18 19 20`

9. Write a C program that receive integer numbers, `n` and `m` from users, where `m<n`. Use loop to display the numbers between 1 to 1000 that are divided by `n` and have the remainder `m`
   Example: `n=11, m=2; 2 13 24 35 46 57 68 79 90 101 112 123 134 145 156 167 178 189 200 211 222 233 244 255 266 277 288 299 310 321 332 343 354 365 376 387 398 409 420 431 442 453 464 475 486 497 508 519 530 541 552 563 574 585 596 607 618 629 640 651 662 673 684 695 706 717 728 739 750 761 772 783 794 805 816 827 838 849 860 871 882 893 904 915 926 937 948 959 970 981 992`

10. Write a C program that simulates a simple guessing game. The program should generate a random number between 1 and 100 (inclusive) and ask the user to guess the number. The user should continue to guess until they correctly guess the number. Note: the program must use a do-while loop for this purpose.

11. Write a C program that calculates and displays the Fibonacci sequence up to `n` terms. Use a do-while loop to repeatedly ask the user for the number of terms they want to generate and then display the corresponding Fibonacci sequence, until users input a negative number.
    Example: `n=8; 0, 1, 1, 2, 3, 5, 8, 13`